- ***-> operations with numpy***
    - -> element wise addition, subtractions
    - -> a + 2 <- this adds 2 to each element in the a array
    - -> you can also -, *, /
        - -> to all of the elements in the array
    - -> you can also create another array
    - -> he has defined a second array called b
    - -> then added two arrays together -> each of their elements are added
    - -> you can ** raise to a power
        - -> each of the elements in the array are raised to the power

- ***-> you can also take the sin of all of the values***
    - -> np.sin(a)
    - -> the sin of all of the elements in an entire array
    - -> these operations can all be done at once
    - -> you can also do a cosine

- ***-> linear algebra***
    - -> functions which you can use on elements
    - -> when you are using matlab
    - -> we are not doing element wise computation -> for example multiplying matrices
    - -> he has defined two matrices
    - -> one is 2x3 and the other is 3x2
    - -> we should end up with a 2x2 matrix at the end
    - -> np.matmul(a,b) <- to times two matrices together
        - -> when you times matrices, they don't have to have the same dimensions
    - -> dot product
        - -> np.identity(3)
        - -> np.linalg.get(c)
    - -> you can also do eigenvalues, the inverse of a matrix
        - -> you can use the scipy documentation for this

- ***-> statistics with numpy***
    - -> min, mean, max
    - -> we have an example array, which we want the minimum of
    - -> np.min(stats)
    - -> np.max(stats)
    - -> stats is the name of the array
    - -> you can also set the axis argument equal to 0 <- this tells it to work with the top row
- -> np.sum(stats) <- the sum of all the elements in the matrix

- ***-> question***

What is the value of b after running the following code?

```
import numpy as np

a = np.array(([1, 2, 3, 4, 5], [6, 7, 8, 9, 10]))
b = np.max(a, axis=1).sum()
```

- options
    - 10
    - 7

- 5
- 15 <- This one